I. J. Computer Network and Information Security, 2019, 2, 21-27
Published Online February 2019 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2019.02.03

An Efficient Steganography Technique for Images using Chaotic Bitstream

Hidayet Ogras
Department of Electrical Education, Batman University, Batman, 72100, Turkey
E-mail: hidayet.ogras@batman.edu.tr

Received: 30 November 2018; Accepted: 14 December 2018; Published: 08 Febuary 2019

***Abstract*—**Steganography is a science of security technique through invisible communication; hiding secret message into cover objects without any suspicion. Protection of the hidden information from an adversary is the main purpose of any steganography system even if the embedding method is obvious. In this paper, an effective image steganography system based on a least significant bit (LSB) algorithm and chaos is proposed. The proposed method contains a spatial domain technique in which the Logistic map is used for generating chaotic bitstream and bitwise XOR operation which is used to create control bit. Control bit determines whether the LSB of a specific pixel is changed or not according to the secret message. This provides direct manipulation over the pixels of the image with a very low precision hence enhances the system security. In this study, gray image is used as secret message and a larger scale image as cover image. Experimental results demonstrate that the proposed method is very efficient to detect LSB replacement in the algorithm. Moreover, the proposed algorithm is highly sensitive to the stego key parameters due to complex structure of chaos which provides high level of security in the whole system and effectively hides and detects the image information.

***Index Terms*—**Image steganography, chaos, logistic map, information hiding, LSB replacement.

I. Introduction

Steganography aims to embed secret information into a cover medium with the purpose of data security, identification or copyright protection so that intruders are not able to detect the messages [1,2]. Steganography distinctly differs from cryptography such that the cryptography deals with changing message data into an unreadable form whereas steganography tends to hide the message data into a cover medium which makes it difficult for an observer to figure out where the message is. Digital media files such as images, audio and video files are used as cover media of steganography [3,4]. Especially image files are very popular for cover medium due to the having great amount of redundant space which is suitable to hide secret information. Steganography methods generally use similar key to hide secret message and generate an output data called stego. Stego should be undetectable from the cover media in order to obscure the communication between the two sides [5]. The most popular and frequently used image steganography technique is LSB substitution [6]. LSB insertion is a simple process for modifying least significant bit of the pixels of the cover image as some or all of the bytes inside the image is changed to a bit of the secret message in spatial domain. Spatial domain means that only the pixel values are considered to directly embed the secret information into some specific value of the image. Two important spatial domain methods exist in steganography: AE-LSB (Adaptive data hiding in edge areas of images with LSB domain) and EA-LSBMR (Edge Adaptive Image Steganography based on LSB Matching Revisited). The first algorithm is variable-sized embedding technique inserting a variable number of secret bits in the pixels of the cover image. The second one is a fix-sized inserting a constant number of bits in all pixels [7]. LSB replacement embeds a secret message into the cover image with message bit. As a result, pixel value is increased or decreased by 1. If the value of a pixel of an image is changed by a value of '1', it affects nothing on the appearance of the image. This makes possible for hiding data in an image especially when the cover file is bigger than the message file. In this study, properties of chaos dynamics such as pseudo-randomness, ergodicity and sensitivity of the system parameters are utilized in order to make hiding message more secure.

The rest of the paper is organized as follows: Section II gives related works about data hiding of steganography. Section III introduces chaotic Logistic map with its dynamical features. The proposed image steganography algorithm is given in detail in Section IV. Section V presents experimental results for the proposed algorithm. Security analyses of the proposed algorithm are given in Section VI. Results for the performance comparison with similar methods are given in Section VII. Finally, Section VIII concludes the paper.

II. Related Works

Dogan [8] has proposed a new data hiding algorithm based on pixel pairs recently. The author used a chaotic map as a PRNG and its output determines whether addition or subtraction process is applied on pixel pairs for data hiding. The proposed algorithm achieves high

payload capacity, good running time and security. Sun [9] introduced another image steganography scheme based on improved Logistic map and DNA sequence. In this study, two secret bits are embedded into the edge pixels and Canny edge algorithm detects edge pixels of the cover image. Valendar at al. [5] presented a new transform domain steganography method based on integer wavelet transform for digital images having high capability for hiding information and high level of security. There are also some studies involving both cryptography and steganography in successive order for data hiding. For instance, Sharma et al. [10] proposed an image hiding method that contains a cryptographic algorithm and a steganography technique using LSB approach. The suggested algorithm fully satisfies the basic factors of information security. Varsha at al. [11] proposed a technique for data hiding using steganography and cryptography. Firstly, text data is encrypted with RSA algorithm and then the encrypted data is embedded into the image file using advanced LSB method. The suggested method demonstrates high level of security.

III. Chaotic Logistic Map

Logistic map is a simple one-dimensional discrete system that exhibit chaos and defined in (1).

$$x_{n+1} = r.x_n(1-x_n) \tag{1}$$

Here, $0 < r \le 4$ is called control parameter of the system and $x_n \in (0,1)$. When $r \in [3.57,4]$, then the map is in chaos state which means x_n is aperiodic, non-convergent and very sensitive to initial value. Logistic map can demonstrate different behaviors, from a stationary when r is close to 0, to a chaotic when r is close to 4. This can be seen from the bifurcation diagram of the map as shown in Fig.1.

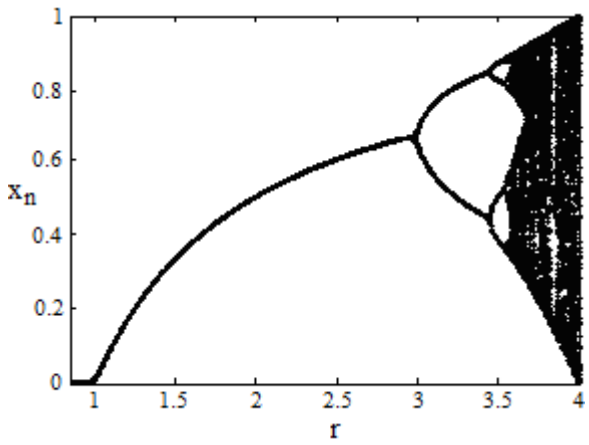


Fig.1. Bifurcation Diagram of the Logistic Map

Fig.2 shows the output of the Logistic map when the system is in chaos state with the parameters of $r = 4$ and $x_0 = 0.123$.

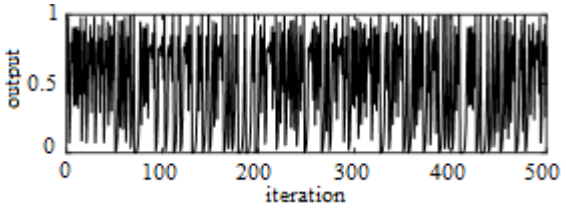


Fig.2. Output of the Chaotic Logistic Map

Logistic map has been widely used in many scientific fields such as steganography [4,5,8,9,12,13] and cryptography [14,15,16,17,18] as a pseudo-random number generator.

IV. The Proposed Algorithm

Image steganography technique is a process of hiding data within the image in a way that prevents the intruders from the detection of the hidden data. This hidden information can be retrieved only through proper decoding method. The proposed method aims to use steganography for an image with another image in spatial domain by changing LSB of the cover image through a control bit. First of all, the Logistic map is used to generate chaotic sequential bitstream by using (2) which is applied to its output.

$$b_n = \begin{cases} 1 \ , & x_n \ge 0.5 \\ 0 \ , & x_n < 0.5 \end{cases} \tag{2}$$

Hence, a bit value ‘1’ or ‘0’ is obtained for each x_n.

A. LSB Method

In a grayscale image each pixel is represented in 8 bits. The last bit in a pixel is called LSB as its value will affect the pixel value only by ‘1’. Therefore, the visual quality of the stego image will be high but the data capacity for the message will be low. In this study, the security of the hidden image data is considered rather than data capacity. In this sense, the advantages of chaos properties are used to make more complex structure in the LSB replacement algorithm during the embedding process. Firstly, all the pixel values of the secret gray image are converted to binary values. For example, if the message image has a size of 100x100, then the total number of bits in this image will be 80,000. Secondly, this number of pixels is selected sequentially from up to down and left to right in the cover image. After that, each bit in the secret data is embedded into the LSB of the selected pixels correspondingly. As a result, the secret data are spread out among the image data. LSB value may not change depending on whether the control bit is same or not with the message bit during the embedding. The embedding process depends on the secret stego keys which are the control parameter and initial condition of the chaotic logistic map. The control bit is obtained by using a simple XOR operation through chaotic bitstream and LSB of the selected pixels. This operation is defined in (3).

$$control\_bit = LSB \oplus \{chaotic\_bitstream\} \quad (3)$$

Notice that this control bit is sensitive to the pixels in the cover image and parameters of the Logistic map. Status of the control bit according to the message bit will define whether the LSB is modified or not. This situation is shown in Table 1.

Table 1. Action List for LSB of the Selected Pixels

Possibilities	LSB	Appropriate action to the LSB	Modified LSB
c_bit = m_bit	0	No change required	0
c_bit ≠ m_bit	1	Decrement	0
c_bit = m_bit	1	No change required	1
c_bit ≠ m_bit	0	Increment	1

Here, *c* and *m* refer control and message bit, respectively. Increment or decrement process depends on the bit value of the LSB before the embedding. For example, if the control bit is not same with the message bit and LSB is '0', then modified LSB will be '1'. Truth table is shown in Table 2. Block diagram for this process is shown in Fig.3.

Table 2. Truth Table for Generating Modified LSB

Message Bit	Control Bit	LSB	Chaotic Bit streams	Modified LSB
0	0	0	0	0
0	1	0	1	1
0	1	1	0	0
0	0	1	1	1
1	0	0	0	1
1	1	0	1	0
1	1	1	0	1
1	0	1	1	0

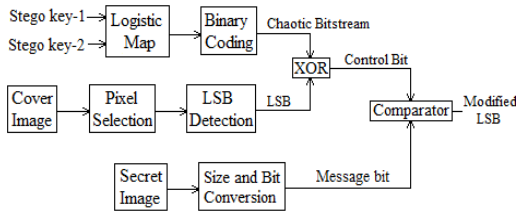


Fig.3. Block Diagram of the Generating Modified LSB

To complete the embedding process, LSB values of the selected pixels in the cover image are replaced with modified ones which yield the stego-pixel values.

B. Message Decoding

For message decoding, image operations and reverse of the encoding process are combined to recover the secret message. The decoding process is easier ve faster than the embedding process. Message bit can be recovered by the same XOR operation with chaotic bitstream and modified LSB as in (4).

$$message\_bit = chaotic\_bit \oplus \mathrm{mod}ified\_LSB \quad (4)$$

The validity of (4) can be checked from the Table 2. Since the Logistic map is used to generate chaotic bitstream, identical stego key parameters must be used at the receiver side to produce same chaotic bitstream for decoding process. Embedded information is secure as long as the system parameters of the Logistic map are unknown. Block diagram of the recovering message bit is shown in Fig.4.

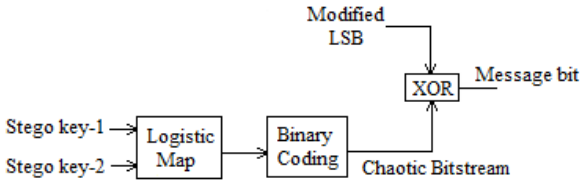


Fig.4. Block Diagram of the Recovering Message Bit

V. Experimental Results

Four test images with different sizes: 'Lena', 'Baboon', 'Peppers' and 'Cameraman' are considered to evaluate the performance of the proposed method. These images are used as gray level cover images and 'Couple' images with different sizes of 64x64, 100x100, 128x128 and 256x256 are used as secret images as shown in Fig.5.

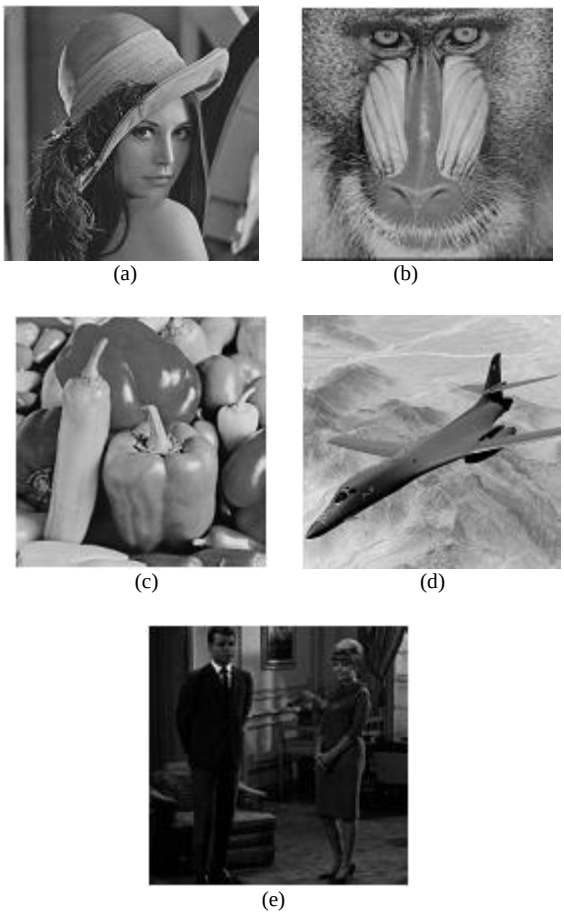
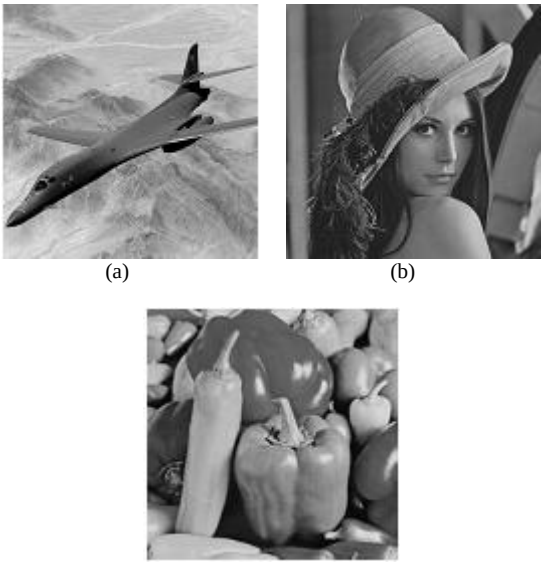


Fig.5. Test images: (a) Lena, (b) Baboon, (c) Pepppers, (d) Airplane, (e) Couple

In order to evaluate the performance of the proposed steganography method, enough number of simulations are tested and theoretical analyses are performed and also

running speed of the algorithm as encoding/decoding rates are calculated on Intel Core i7 3.4 Ghz CPU with 4 GB RAM by using MATLAB 2015a. The average execution time for the results can be found in Table 4. For analyzing the performance and security of the proposed method, 'Couple' image (100x100) are hidden within the cover images: 'Airplane' (1024x1024), 'Lena' (512x512), 'Peppers' (300x300). Resultant stego-images are shown in Fig.6. Correlation coefficient, entropy, Peak Signal-to-Noise Ratio (PSNR) and Image Fidelity (IF) are considered for theoretical analyses.



(a) (b)

(c)

Fig.6. Stego images: (a) Airplane, (b) Lena, (c) Peppers

PSNR computes the peak signal-to-noise ratio, in decibels between two images. This ratio is generally used as a quality measurement between two images and it is defined in (5).

$$PSNR = 10\log_{10}\frac{255^2}{MSE} \tag{5}$$

Mean square error, shortly MSE, represents the difference between the stego image and cover image with a size of *HxW*. MSE is defined in (6).

$$MSE = \frac{1}{H \times W}\sum_{i=1}^{H}\sum_{j=1}^{W}(C_{i,j} - S_{i,j})^2 \tag{6}$$

Here, $C_{i,j}$ and $S_{i,j}$ represent cover image and stego image, respectively. MSE shows the quantitative representation of the error that occurs in the stego image with respect to the cover image. The lower value of the MSE means lower error between two images and higher performance of the algorithm. Bigger PSNR value means that a stego image is very similar to its original image according to the visual quality and having high embedding efficiency. Generally, if the PSNR value is higher than the 30 dB, then it is very hard to distinguish the distortion by human eyes [19]. Entropy is a major factor in terms of robustness. It shows the probability of occurrence of pixels in stego image should be equal to the original image. The entropy of an information source with a length of *N* is determined in (7).

$$H(X) = -\sum_{i=0}^{N-1} p(x_i).\log_2 p(x_i) \tag{7}$$

$H(X)$ and $p(x_i)$ represent the information entropy in bits and the probability of symbol x_i, respectively [14]. Image Fidelity (IF), is the perceptual similarity between cover and stego images before and after the stego processing. Correlation coefficient parameter is used to find the linear correlation between cover and stego images. This value must be between -1 and 1. 1 indicates a strong positive relationship; -1 indicates a strong negative relationship and 0 indicates no relationship at all. Numerical results are listed in Table 3.

Table 3. Experimental Results for the Proposed Method

Size	Images	Coeff.	PSNR	IF	Entropy
1024 x 1024	Cover Airplane	0.999504	65.277	0.999923	7.20493
	Stego Airplane				7.20646
512 x 512	Cover Lena	0.998817	58.626	0.999650	7.22978
	Stego Lena				7.22969
300 x 300	Cover Peppers	0.999847	53.713	0.998907	7.56256
	Stego Peppers				7.56321

According to the above results, it is obvious that the obtained results are satisfactory and confirm that the proposed stego technique has high visual quality and the embedding process is working well.

VI. Security Analyses

A. Key Space Analysis

The key space size is the total number of different stego keys that can be used in a steganography system. According to the IEEE floating-point standard [20], the computational precision of the 64-bit double precision number is about 10^{15}. In the proposed scheme, stego keys are r and x_0. They have floating point values. Hence, the total number of possible stego keys can be given in (8).

$$stego\_key = 10^{15\times 2} \approx 2^{100} \tag{8}$$

This value is sufficiently large to resist brute-force attack. To give a sense of what this number mean, a powerful computer that could check 304,510 MIPS (AMD Ryzen 7 Processor) of combinations per second, it takes too many years to crack the system as in (9).

$$\frac{2^{100}}{304510\times 10^6 \times 60 \times 60 \times 24 \times 365} = 1.32\times 10^{11} \tag{9}$$

According to the result, it is practically not applicable so the proposed method is said to be secure for brute-force attacks.

B. Key Sensitivity Analysis

Key sensitivity can be observed in an aspect: if a tiny difference exists in stego key in receiver side, then the stego image could not be decoded correctly. For this case, one of the test images, 'Airplane' (1024x1024) is selected as a cover image and a secret image of 'Lena' (256x256) is used to hide with a randomly chosen stego Key-1 as $r = 3.999999999$, $x_0 = 0.123456789$. Key-1 is used to hide secret 'Lena' image within 'Airplane' image using the proposed method. Then a slight change is applied to the one of the key parameters while other remains same and then tries to decode the 'Lena' image in the receiver side. These are: Key-2 is $r = 3.99999998$, $x_0 = 0.123456789$; Key-3 is $r = 3.999999999$, $x_0 = 0.123456788$. Key-1 creates Stego Image-1; Key-1, Key-2 and Key-3 are used to decode secret message from the same stego image. Fig. 7 shows the visual results for the key sensitivity analysis.

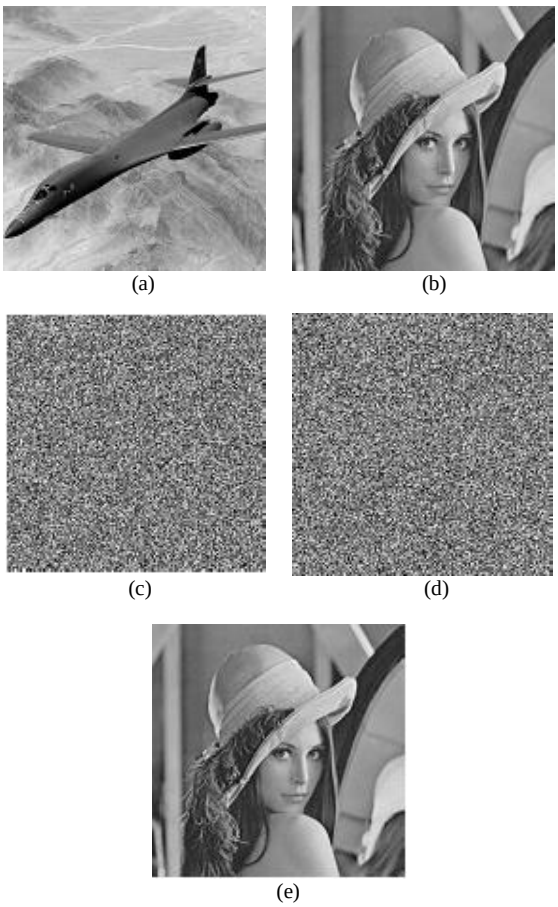


Fig.7. Key Sensitivity Result: (a) Stego Airplane Image Generated with Key-1, (b) Secret Lena Image, (c) Wrong Decoded Image with Key-2, (d) Wrong Decoded Image with Key-3, (e) Correct Decoded Image with Key-1

The identical key that is used for embedding the secret image, can only decode the hidden image correctly. As a result, the proposed stegonagraphic system is quite sensitive to both stego keys which improves the security of the hidden image.

C. Histogram Analysis

In image processing, histogram is used to show the number of pixels in an image at each different intensity values and gives statistical information about the image. The ideal histogram of a stego image should be similar to the original cover image. The histograms of the cover and stego 'Airplane' images are shown in Fig.8.

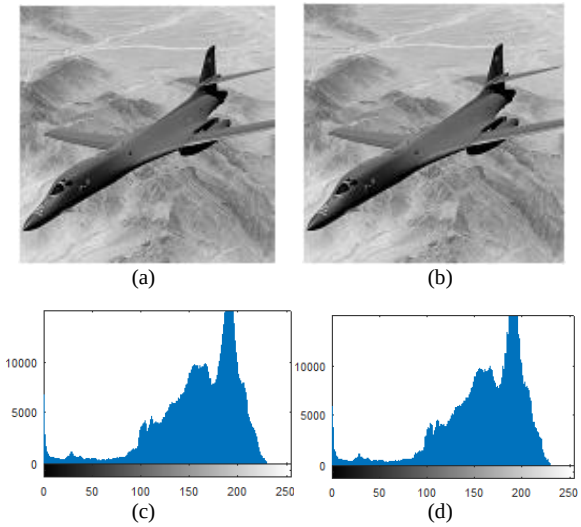


Fig.8. Histogram Results: (a) Cover Airplane Image, (b) Stego Airplane Image, (c) Histogram of (a), (d) Histogram of (b)

It is obvious that histogram results for cover and stego images are almost same that results minimum distortion in the image. The histograms of the hidden and decoded 'Lena' images are shown in Fig.9.

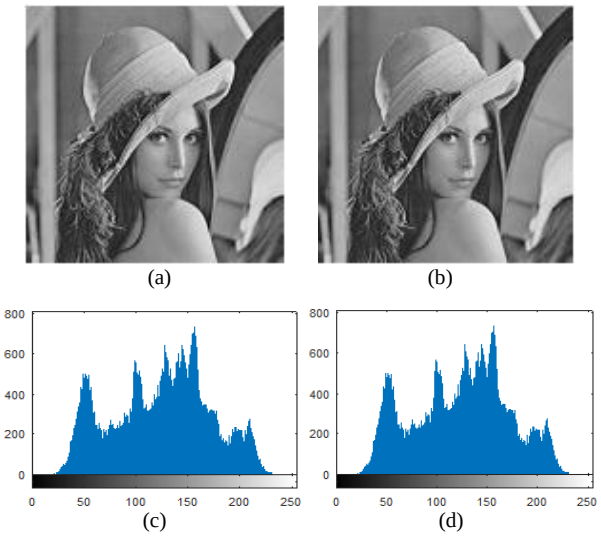
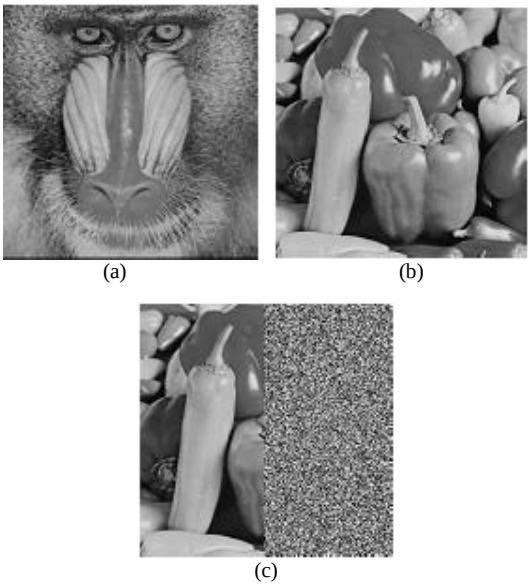


Fig.9. Histogram Results: (a) Hidden Lena Image, (b) Decoded Lena Image, (c) Histogram of (a), (d) Histogram of (b)

For the application of the proposed method, the size of the cover image must be at least $\sqrt{8}$ greater than the size of the secret image in order to embed the secret image smoothly. For instance, a secret image having 256x256 in size, can not be embedded completely in a cover image

with a size of 512x512 as shown in Fig.10.



(a) (b)

(c)

Fig.10. (a) Stego Baboon Image (512x512), (b) Secret Peppers Image (256x256), (c) Decoded Peppers Image

Average running time for data hiding and data decoding processes are given in Table 4.

Table 4. Average Running Time for Data Hiding and Data Decoding of the Proposed Algorithm

Test Image	Secret Couple Image	Data Hiding (sec)	Data Decoding (sec)
Airplane (1024x1024)	32x32	0.0164	0.0095
	64x64	0.0207	0.0198
	100x100	0.0438	0.0310
	128x128	0.0591	0.0477
	256x256	0.0830	0.0692

VII. Performance Comparison

For the performance evaluation of the proposed scheme, different size of secret 'Couple' images and various cover images are used to calculate PSNR and IF values. Then the obtained results are compared with other algorithm [21]. Results for PSNR and IF are listed in Table 5 and Table 6, respectively.

Table 5. Performance Comparison of PSNR Results

Cover Images	Secret Couple Image	[21] AE-LSB PSNR	[21] EA-LSBMR PSNR	Proposed Scheme PSNR
Lena (512x512)	32x32	60.03	70.35	64.82
	64x64	54.42	64.41	61.74
	100x100	50.33	60.51	59.33
	128x128	48.32	58.35	56.87
Baboon (512x512)	32x32	57.55	70.52	58.71
	64x64	51.27	64.46	54.28
	100x100	47.19	60.59	52.37
	128x128	45.20	58.41	51.96
Peppers (512x512)	32x32	59.43	69.71	62.19
	64x64	54.52	63.78	59.56
	100x100	50.25	59.86	57.14
	128x128	48.04	57.70	56.27

Table 6. Performance Comparison of IF Results

Cover Images	Secret Couple Image	[21] AE-LSB IF	[21] EA-LSBMR IF	Proposed Scheme IF
Lena (512x512)	32x32	0.9998	1.0000	0.999913
	64x64	0.9991	0.9999	0.999825
	100x100	0.9978	0.9998	0.999658
	128x128	0.9965	0.9997	0.999494
Baboon (512x512)	32x32	0.9996	1.0000	0.999830
	64x64	0.9981	0.9999	0.998378
	100x100	0.9952	0.9998	0.998106
	128x128	0.9924	0.9996	0.998523
Peppers (512x512)	32x32	0.9998	1.0000	0.999788
	64x64	0.9992	0.9999	0.999546
	100x100	0.9979	0.9998	0.999477
	128x128	0.9966	0.9996	0.999328

It is concluded that the proposed algorithm shows better performance than the AE-LSB algorithm but it has lower results than the EA-LSBMR. When the size of secret messages increase, then the PSNR and IF values decrease for all algorithms. Another similar steganography algorithm that is recently proposed [22] is compared with the proposed algorithm. Results are listed in Table 7.

Table 7. PSNR Values Versus of [22]

Cover Images	Secret Image	Number of Hidden Bits	Ref.[22]	Proposed Method
Lena (512x512)	Flower (192x192)	294912	43.60	54.18
Baboon (512x512)	Flower (128x128)	131072	51.13	53.41

According to the Table 7 results, the proposed method has better PSNR values than the results of [22]. Table 8 shows the amount of time required to embed secret image into the cover image for [9] and the proposed method.

Table 8. Average Time (sec) Required for Embedding Secret Image

Cover Images	Ref. [9]			Proposed Method		
	Secret Image Size			Secret Image Size		
	32x32	64x64	81x81	32x32	64x64	81x81
512x512	0.146	0.156	0.166	0.0160	0.021	0.032

Table 8 shows that the proposed method embeds the same amount of information into the equal size of cover image much faster than the results of [9].

VIII. Conclusions

In this study, an image hiding algorithm based on chaos is proposed. Important features of the chaotic dynamics are utilized to improve the security of the hidden image and create a structure of key sensivity for both hiding and decoding processes in the proposed algorithm. A small change in the stego key alters chaotic bitstream significantly that effects control bit in the algorithm hence improves the security of the decoding process. LSB method is used to hide secret image and a very simple XOR operation is used to decode the secret

image data effectively. Experimental results show that the proposed steganography method is very efficient to embed secret image and detect secret data without error. Visual and numerical results confirm high visual quality, good security and good running time of the proposed scheme. The hardware implementation of the proposed algorithm is possible direction for my future works.

REFERENCES

[1] A. A. J. Altaay, S. B. Sahib and M. Zamani, "An introduction to image Steganography techniques," *International Conference on Advanced Computer Science Applications and Technologies*, Malaysia, November 2012, pp. 122–126.

[2] V. L. Reddy, "Novel chaos based Steganography for Images using matrix encoding and cat mapping techniques," *Information Security and Computer Fraud*, vol. 3(1), pp. 8-14, 2015.

[3] J. Fridrich, M. Goljan and R. Du, "Reliable detection of LSB Steganography in Color and Grayscale images," *Proceedings of the 2001 Workshop on Multimedia and security: new challenges*, Canada, October 2001, pp. 27-30.

[4] K. Satish, T. Jayakar, C. Tobin, K. Madhavi and K. Murali, "Chaos based spread spectrum image Steganography," *IEEE Transactions on Consumer Electronics*, vol. 50(2), pp. 587-590, 2004.

[5] M. Y. Valandar, P. Ayubi and M. J. Barani, "A new transform domain steganography based on modified logistic chaotic map for color images," *Journal of Information Security and Applications Transactions on Consumer Electronics*, vol. 34(2), pp. 142-151, 2017.

[6] A. Pradhan, A. K. Sahu, G. Swain and K. R. Sekhar, "Performance evaluation parameters of image Steganography techniques," *International Conference on Research Advances in Integrated Navigation Systems, India, December 2016, pp. 1-8.*

[7] C.-H. Yang, C.-Y. Weng, S.-J. Wang and H.-M. Sun, "Adaptive data hiding in edge areas of images with spatial LSB domain systems," *IEEE Transactions on Information Forensics and Security*, vol. 3(3), pp. 488-497, 2008.

[8] S. Dogan, "A new approach for data hiding based on pixel pairs and chaotic map," *I. J. of Computer Network and Information Security*, vol. 10(1), pp. 1-9, 2018.

[9] S. Sun, "A novel secure image steganography using improved Logistic map and DNA techniques," *J. of Internet Technology*, vol. 18(3), pp. 647-652, 2017.

[10] H. Sharma, M. Arya and D. Goyal, "Secure image hiding algorithm using Cryptography and Steganography," *IOSR Journal of Computer Engineering*, vol. 13(5), pp. 1-6, 2013.

[11] Varsha, R. S. Chhillar, "Data hiding using Steganography and Cryptography," *I. J. of Computer Science and Mobile Computing*, vol. 4(4), pp. 802-805, 2015.

[12] M. Ulker and B. Arslan, "A novel secure model: Image steganography with logistic map and secret key," 6th International Symposium on Digital Forensic and Security, Turkey, May 2018.

[13] S. V. K. Yadav, S. Batham, "A novel approach of bulk data hiding using text Steganography," *Procedia Computer Science*, vol. 57, pp. 1401-1410, 2015.

[14] H. Ogras and M. Turk, "A Robust chaos-based image cryptosystem with an improved key generator and plain image sensitivity mechanism," *J. of Information Security*, vol. 8, pp. 23-41, 2017.

[15] L. Quan, L. Pei-Yue, Z. Ming-Chao, S. Yong-Xin and Y. Huai-Jiang, "A Novel image encryption algorithm based on chaos maps with Markov properties," *Communications in Nonlinear Science and Numerical Simulation*, vol. 20(2), pp. 506-515, 2015.

[16] N. K. Pareek, V. Patidar and K. K. Sud, "Image encryption using chaotic logistic map," *Image and Vision Computing*, vol. 24(9), pp. 926-934, 2006.

[17] P. J. Rani and S. D. Bhavani, "Symmetric encryption using Logistic map," *1st International Conference on Recent Advances in Information Technology, India, March 2012.*

[18] H. Ogras and M. Turk, "An efficient method to improve the Logistic map: Design and Implementation," Third International Conference on Electrical, Electronics, Computer Engineering and their Applications, Lebanon, May 2016.

[19] H. Tarrah and S. Mirzakuchaki, "Efficient steganography scheme based on Logistic map and DWT-SVD," *I. J. of Computer Applications*, vol. 164(8), pp. 8-11, 2017.

[20] J.-X. Chen, Z.-L. Zhu, C. Fu, H. Yu and L.-B. Zhang, "A fast chaos-based image encryption scheme with a dynamic state variables selection mechanism," *Communications in Nonlinear Science and Numerical Simulation*, vol. 20(3), pp. 846-860, 2015.

[21] D. Battikh, S. El Assad, B. Bakhache, O. Deforges and M. Khalil, "Chaos-based spatial steganography sytems for images," I. J. of Chaotic Computing, vol. 3(1), pp. 36-44, 2014.

[22] S. A. Al-Taweel, M. H. Al-Hada, A. M. Nasser, "Image in image Steganography Technique based on Arnold Transform and LSB Algorithms," *I. J. of Computer Applications*, vol. 181(10), pp. 32-39, 2018.

Authors' Profiles

Hidayet Ogras received his Ph.D degree in Electrical and Electronics Engineering from the University of Firat, Elazig, Turkey, in 2017. He is currently a research assistant in Batman University and his research interests cover chaos based cryptography and steganography. He is also interested in secure communication systems.

How to cite this paper: Hidayet Ogras,"An Efficient Steganography Technique for Images using Chaotic Bitstream", International Journal of Computer Network and Information Security(IJCNIS), Vol.11, No.2, pp.21-27, 2019.DOI: 10.5815/ijcnis.2019.02.03